Supplement of Sympathetic axonal sprouting induces changes in macrophage populations and protects against pancreatic cancer

Guillot et al.1

Statistical analysis of tumor growth with Bioluminescence data. We relied on the population approach with a non-linear mixed effect model in the Bayesian paradigm to set the model we fit to the data.

1 see the original paper

Keywords: Bayesian statistics; Non-linear mixed models; Gompertz growth model

Table of Contents

[1 Import the data 2](#_Toc98140810)

[2 Gompertz model 3](#_Toc98140811)

[2.1 Examples of Gompertz curves when the value of the plateau changes 4](#_Toc98140812)

[2.2 Examples of Gompertz curves when the value of the speed changes 5](#_Toc98140813)

[3 Least square adjustement to the Gompertz model 6](#_Toc98140814)

[3.1 Principles 6](#_Toc98140815)

[3.2 Numerical results 6](#_Toc98140816)

[3.3 Graphical representation 7](#_Toc98140817)

[4 The Bayesian model 14](#_Toc98140818)

[5 Fitting the Bayesian model 17](#_Toc98140819)

[5.1 Remove the influence of bad measure 17](#_Toc98140820)

[5.2 Samples from the posterior distributions 17](#_Toc98140821)

[5.3 Reproduce Figures 9E and 9F of the paper 18](#_Toc98140822)

[5.4 Summarize the posterior samples 19](#_Toc98140823)

[5.4.1 PK4A-Luc Syngeneic 19](#_Toc98140824)

[5.4.2 R211-Luc Syneneic 21](#_Toc98140825)

[5.4.3 R211-Luc Athymic nude 22](#_Toc98140826)

[5.5 Represent the posterior distibutions 23](#_Toc98140827)

[5.5.1 Reproduce Figure 9G 23](#_Toc98140828)

[5.5.2 Reproduce Figure 9H 24](#_Toc98140829)

[5.5.3 Reproduce Figure 9I 25](#_Toc98140830)

[6 References 27](#_Toc98140831)

# 1 Import the data

Data after syngeneic orthotopic transplantation of PK4A-Luc cells:

expe1 <- read\_xlsx(here::here("analysis/data/raw\_data/guillot-source-data OK.xlsx"),  
 sheet = "Figure 9",  
 range = "Q4:X15")  
expe1 <- long\_log\_format(expe1)

Data after syngeneic orthotopic transplantation of R211-Luc cells:

expe2 <- read\_xlsx(here::here("analysis/data/raw\_data/guillot-source-data OK.xlsx"),  
 sheet = "Figure 9",  
 range = "Q21:V28")  
expe2 <- long\_log\_format(expe2)

Data after transplantation of R211-Luc cells in athymic nude mice:

expe3 <- read\_xlsx(here::here("analysis/data/raw\_data/guillot-source-data OK.xlsx"),  
 sheet = "Figure 9",  
 range = "Q36:V47")  
expe3 <- long\_log\_format(expe3)

Each row of these three datasets is composed of:

* id: a unique ID of the mouse within the experiment
* type: the treatment received by the mouse
* log\_y0: the log-luminescence at time
* tps: time of the observed biolumescence (in days)
* log\_y: logarithm of the observed biolumescence.

# 2 Gompertz model

The Gompertz model (Hartung et al., 2014) assumes that , the size of the tumor at time can be explained as

where:

* is the value of the “plateau”, that is to say the limit of when stabilized,
* is the value at time and
* is a parameter that tunes the approach speed of to the plateau.

The parameters and prescribe the shape of the growth curves, see the examples below, where we let the value of one of these parameters varies whereas the other are fixed to the same value.

## 2.1 Examples of Gompertz curves when the value of the plateau changes

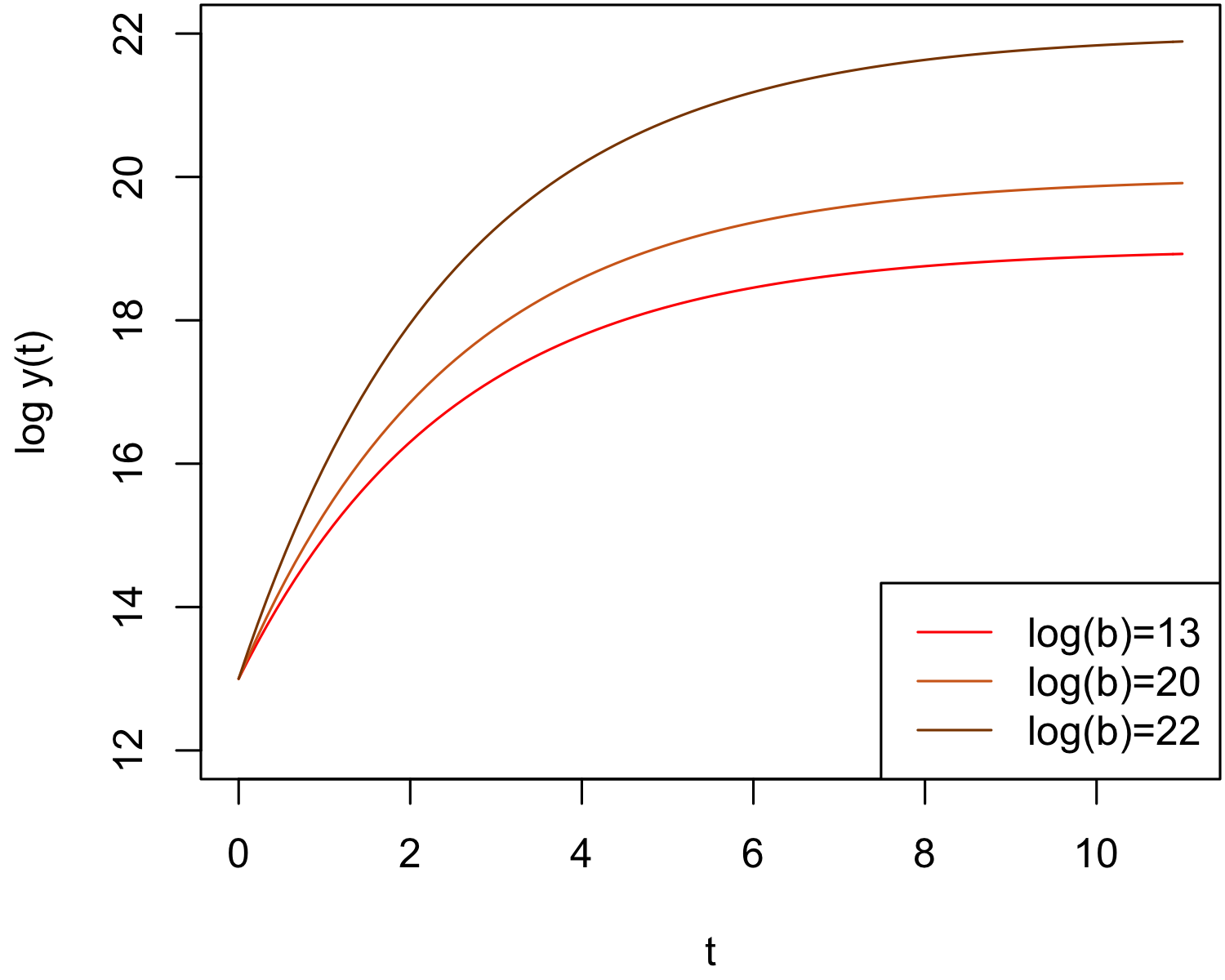


Figure 2.1: Variability of Gompertz growth curves when changes

## 2.2 Examples of Gompertz curves when the value of the speed changes

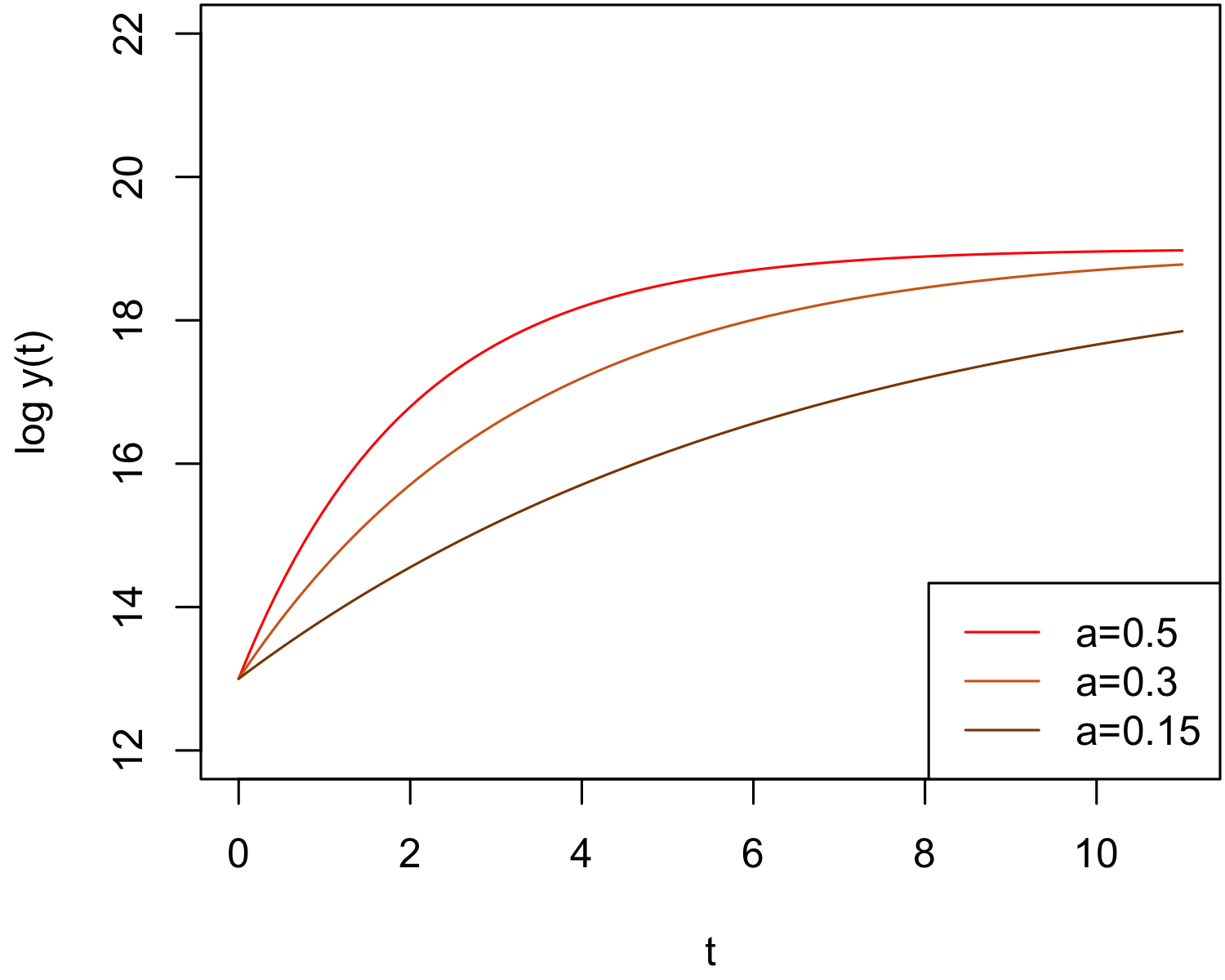


Figure 2.2: Variability of Gompertz growth curves when changes

# 3 Least square adjustement to the Gompertz model

## 3.1 Principles

For the -th individual (mouse) of an experiment, we have to estimate the unknown parameter . We drop the data afte time days. We begin the statistical study with least square estimates that minimize the following criterion:

All were set to , except the following:

* in the 1st experiment, at time , where to avoid fitting growth curve at time where we observed necrosis or hypoxia,
* in the 2nd experiment, at time , where because of bad luminescence measure at this time.

The least square estimates are not our final estimates; we expect them be overfitted. Indeed, for each individual in the sample, we have a five measure to recover three unknown paramters.

## 3.2 Numerical results

par\_LS <- least\_square(expe1 %>% filter(tps <= 8),   
 lambda = c(1, 1, 1, 1, 9))  
par\_LS

| id | type | log\_b | z0 | a |
| --- | --- | --- | --- | --- |
| 1 | AA | 17.14 | 12.87 | 0.3166 |
| 2 | AA | 41.56 | 15.74 | 0.0309 |
| 3 | AA | 18.86 | 15.31 | 0.4439 |
| 4 | AA | 24.53 | 14.42 | 0.0781 |
| 5 | AA | 20.27 | 14.87 | 0.1849 |
| 6 | AA | 17.57 | 15.58 | 0.5204 |
| 7 | AA | 18.22 | 15.46 | 0.4159 |
| 8 | 6-OHDA | 32.99 | 12.51 | 0.0533 |
| 9 | 6-OHDA | 81.76 | 12.54 | 0.0148 |
| 10 | 6-OHDA | 22.31 | 12.67 | 0.1791 |
| 11 | 6-OHDA | 20.89 | 14.20 | 0.2653 |

par\_LS2 <- least\_square(expe2, lambda = c(9, 1, 1, 1, 1))  
par\_LS2

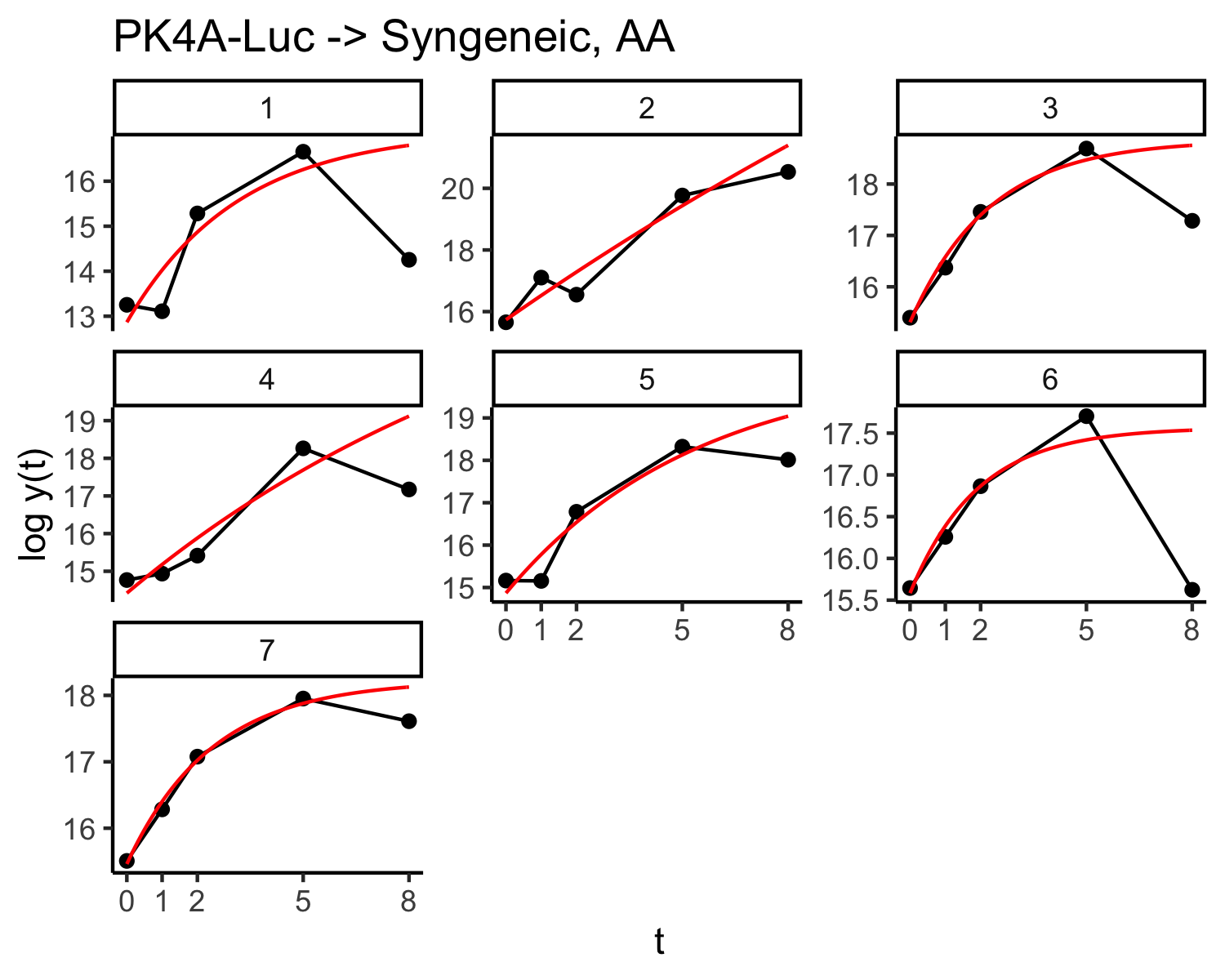
| id | type | log\_b | z0 | a |
| --- | --- | --- | --- | --- |
| 1 | AA | 14.22 | 11.613 | 0.2952 |
| 2 | AA | 15.24 | 10.607 | 0.2763 |
| 3 | AA | 16.34 | 6.970 | 0.1328 |
| 4 | 6-OHDA | 31.47 | 9.078 | 0.0487 |
| 5 | 6-OHDA | 37.07 | 10.433 | 0.0292 |
| 6 | 6-OHDA | 22.91 | 9.158 | 0.0548 |
| 7 | 6-OHDA | 49.16 | 9.688 | 0.0206 |

par\_LS3 <- least\_square(expe3, lambda = c(1, 1, 1, 1, 1))  
par\_LS3

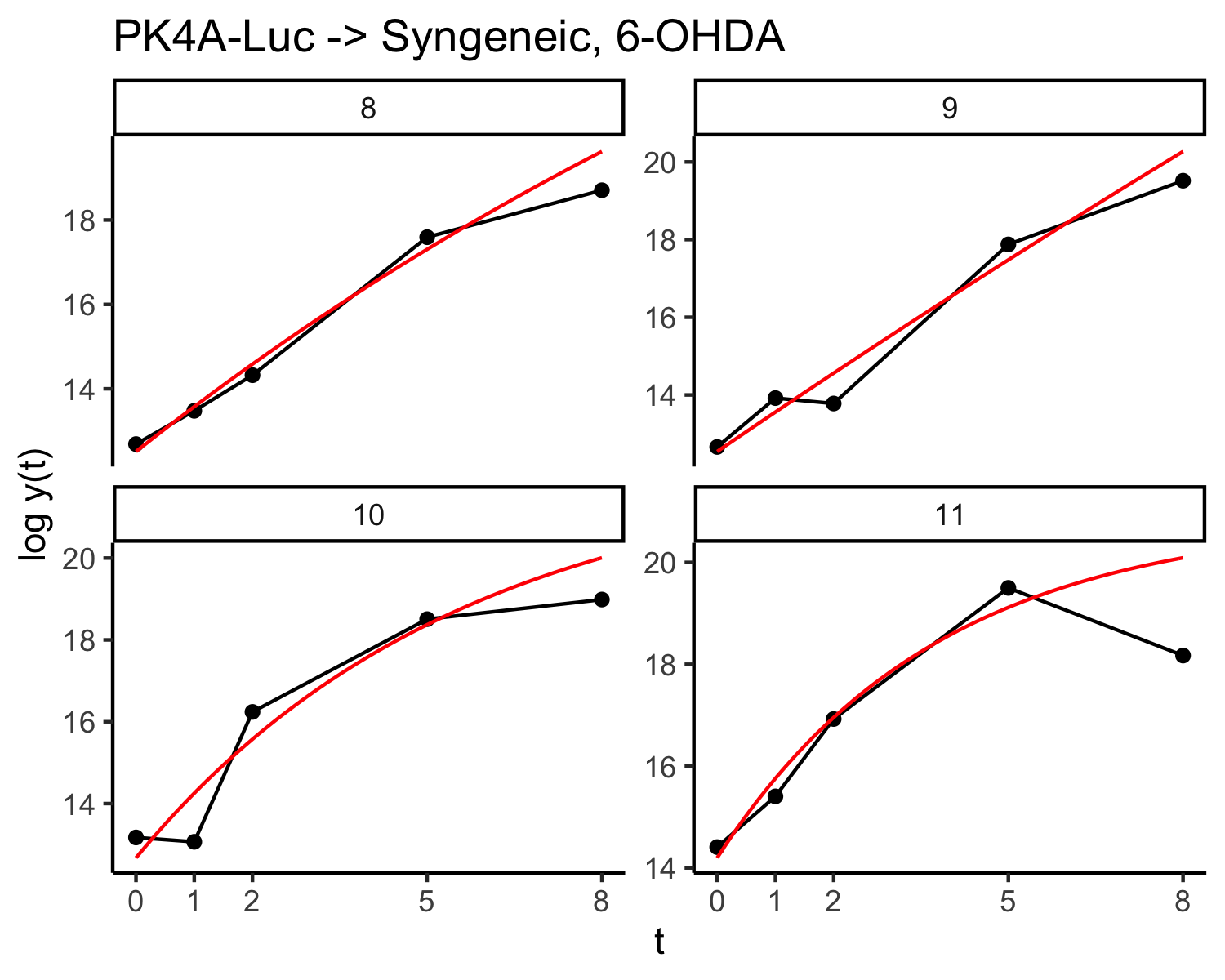
| id | type | log\_b | z0 | a |
| --- | --- | --- | --- | --- |
| 1 | AA | 15.21 | 11.28 | 0.1656 |
| 2 | AA | 18.93 | 12.23 | 0.2105 |
| 3 | AA | 18.57 | 13.49 | 0.4334 |
| 4 | AA | 18.81 | 12.35 | 0.6209 |
| 5 | 6-OHDA | 27.02 | 12.58 | 0.0262 |
| 6 | 6-OHDA | 28.57 | 13.65 | 0.0364 |
| 7 | 6-OHDA | 19.84 | 15.20 | 0.2110 |
| 8 | 6-OHDA | 77.83 | 13.56 | 0.0093 |
| 9 | 6-OHDA | 18.20 | 12.53 | 0.3201 |
| 10 | 6-OHDA | 13.71 | 10.65 | 6.6127 |
| 11 | 6-OHDA | 18.88 | 12.44 | 0.5055 |

## 3.3 Graphical representation

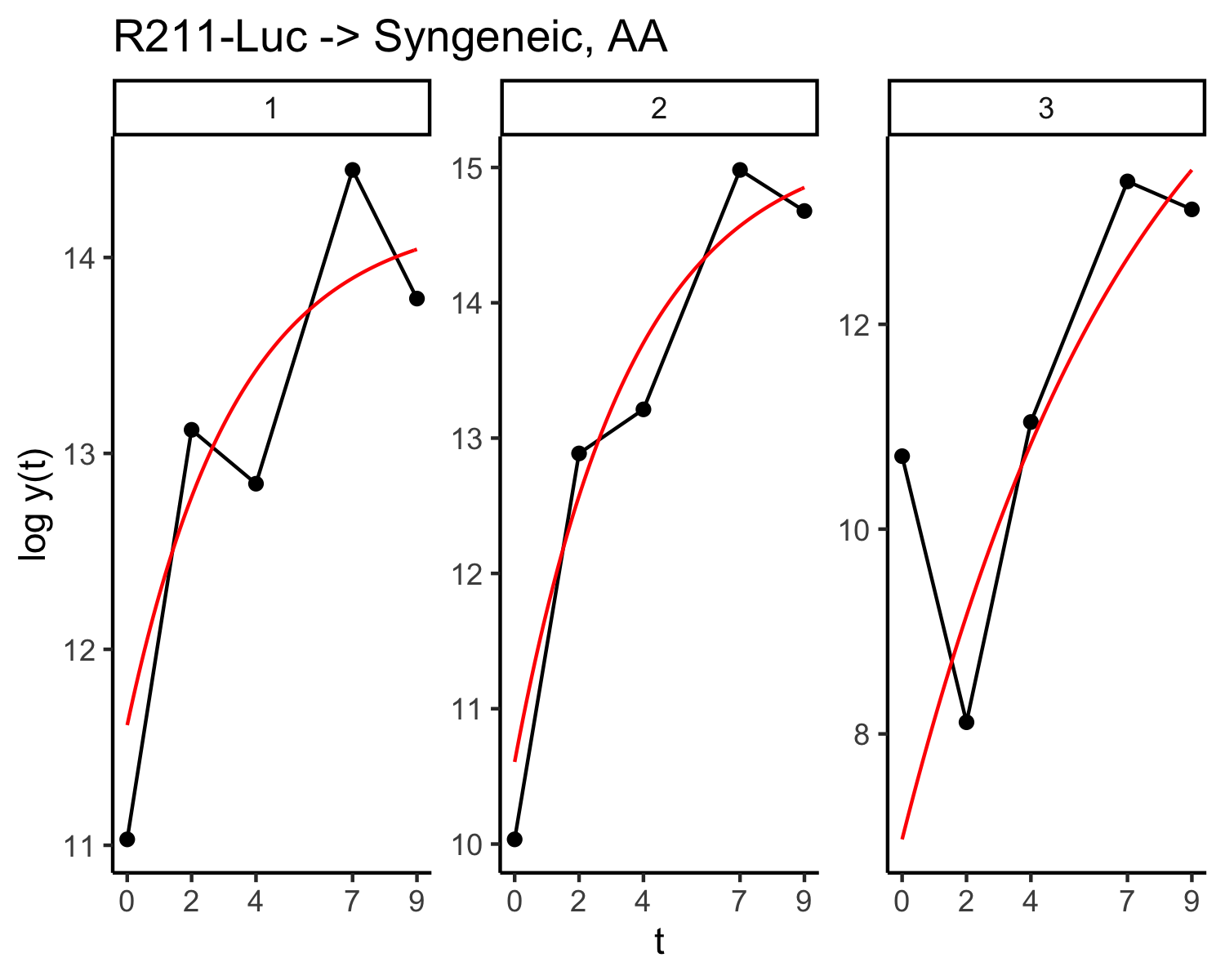
par(mar = c(4, 4, 0, 0) + 0.1)  
plot\_LS(expe1 %>% filter(tps <= 8, type == "AA"),   
 par\_LS %>% filter(type == "AA")) +   
 labs(title = "PK4A-Luc -> Syngeneic, AA")



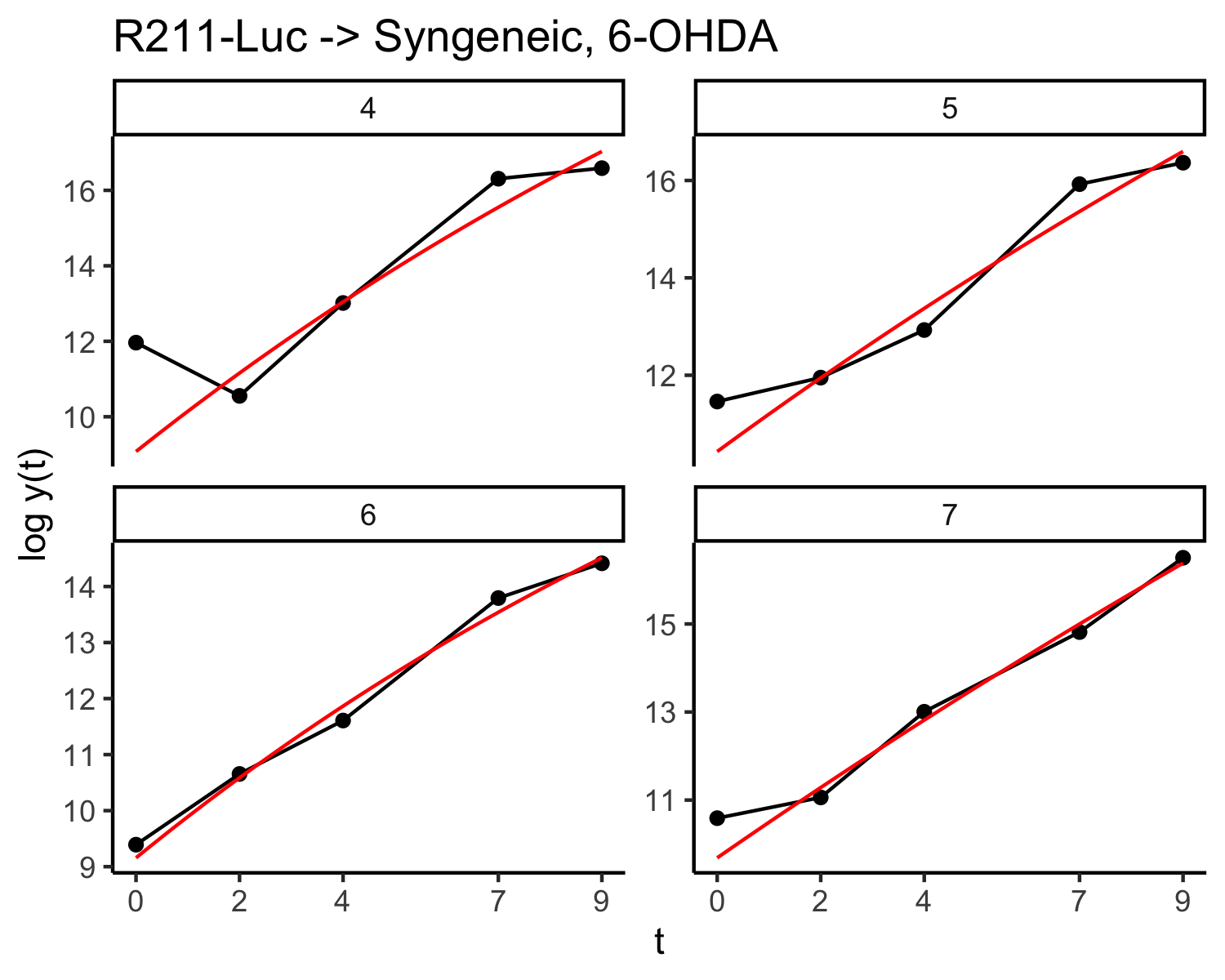
plot\_LS(expe1 %>% filter(tps <= 8, type == "6-OHDA"),   
 par\_LS %>% filter(type == "6-OHDA")) +   
 labs(title = "PK4A-Luc -> Syngeneic, 6-OHDA")



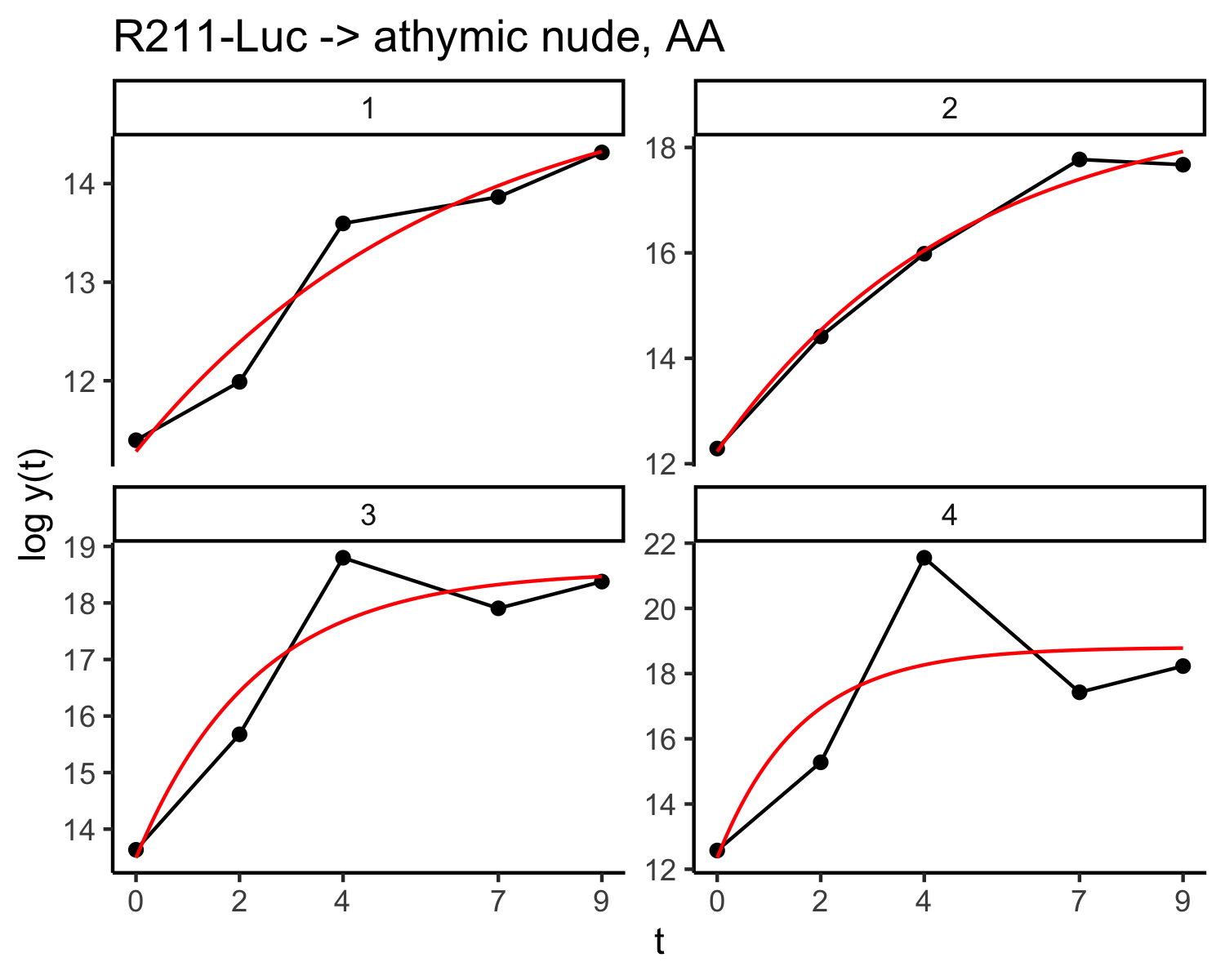
plot\_LS(expe2 %>% filter(type == "AA"),   
 par\_LS2 %>% filter(type == "AA")) +   
 labs(title = "R211-Luc -> Syngeneic, AA")



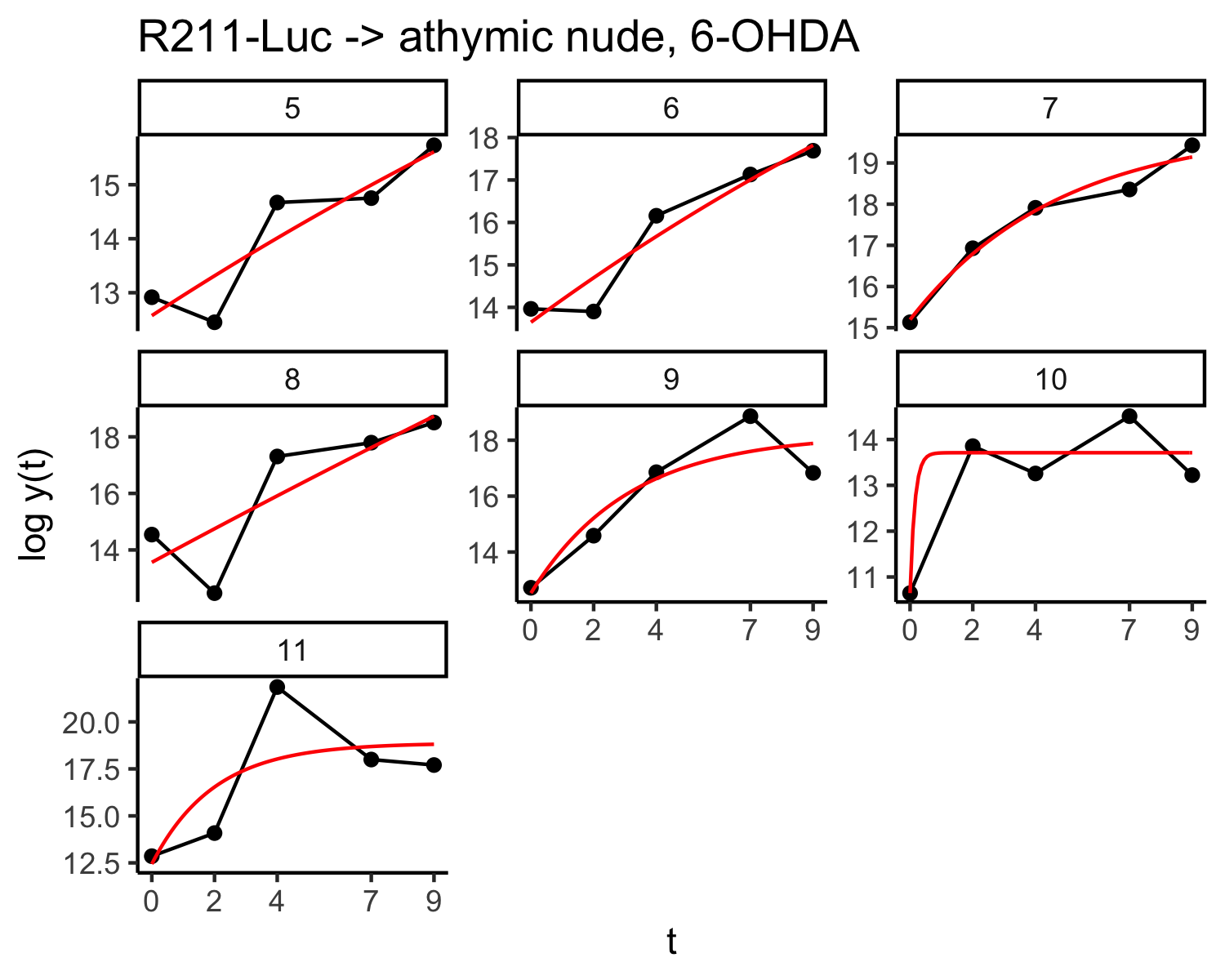
plot\_LS(expe2 %>% filter(type == "6-OHDA"),   
 par\_LS2 %>% filter(type == "6-OHDA")) +   
 labs(title = "R211-Luc -> Syngeneic, 6-OHDA")



plot\_LS(expe3 %>% filter(type == "AA"),   
 par\_LS3 %>% filter(type == "AA")) +   
 labs(title = "R211-Luc -> athymic nude, AA")



plot\_LS(expe3 %>% filter(type == "6-OHDA"),   
 par\_LS3 %>% filter(type == "6-OHDA")) +   
 labs(title = "R211-Luc -> athymic nude, 6-OHDA")



# 4 The Bayesian model

To fit the Gompertz growth curve to the data, we consider a Gaussian additive noise on , which is the logarithm of the bioluminescence measure. More precisely, we assume that the difference between and the value at time of the Gompertz curve is distributed according to a Gaussian centered at . Its variance depends of a -factor which allows us to reduce the influence of a few aberrent data points in the procedure. Thus, except for a few pairs , .

To explain the growth curve at the individual level, we introduce a non-linear mixed effect model (Lavielle, 2014). In the Bayesian paradigm, such models are multilevel or hierarchical models (Gelman & Hill, 2006):

* At the measure level, we model the log-bioluminescence data with a Gompertz growth curve specific to each individual (mouse) in the sample.
* At the group level (either AA or 6-OHDA), we model how the shape of an individual Gompertz growth curve is drawn from priors.
* At the global, we model how other parameters (such as the initial value) are drawn from priors.

At the **measure level**, we have

where:

* is an individual (mouse) ID
* is the -th time of measure
* is the log-bioluminescence measure at time on the -th individual
* is the -th individual parameter of the Gompertz curve
* is the variance of the measurement error
* is a multiplicative factor equal to except for a few pairs
* is a standardized error drawn from the Gaussian .

For each individual, we have a -dimensional parameter to estimate based on the bioluminescence measures.

At the **group level** (the two groups correspond to the two treatments, either AA or 6-OHDA), we set the following prior. For the **plateau**, we have set

where

* is a group ID,
* is an individual ID,
* is the group to which belong the -th individual,
* is the average value of the plateau over group ,
* is the standard deviation of the plateau over group ,
* are prior information about the value of ,
* is prior information about the value of ,
* is a Gaussian distribution,
* is an inverse Gamma distribution.

Hyperparameters , and were set to cover values that have meanings for the problem at hand. In particular, due to the few data points at the individual level, the log-likelihood has many modes. The prior and the chosen values of the hyperparameters allows us to discard uninteresting modes.

For the **speed** to which the curve tends to its plateau, we use the same kind of hierarchical prior, but values of the speed are constrainted to be positive. We set

where

* is a group ID,
* is an individual ID,
* is the group to which belong the -th individual,
* is the average value of the speed over group ,
* is the standard deviation of the speed over group ,
* are prior information about the value of ,
* is prior information about the value of ,
* is a Gaussian distribution constrained to be positive,
* is an inverse Gamma distribution.

**Remark.** The prior information we set with the hyperparameters regarding the average shape of the growth curve at the group level is exactly the same for both groups (AA and 6\_OHDA).

At the **global level**, we set the following prior. We set the following prior on , the error variance:

where

* is the prior density,
* means proportial to, up to number that does not depend on ,
* is a prior bound.

Regarding the **initial values** , we rely on a hierarchical prior that does not distinguish between both groups (AA or 6-OHDA) since we do not expect any difference here. Thus

where

* is an individual ID,
* is the average of the initial value over the whole population,
* is the standard deviation of the initial value over the population,
* are prior information about the value of ,
* is prior information about the value of ,
* is a Gaussian distribution constrained to be positive,
* is an inverse Gamma distribution.

# 5 Fitting the Bayesian model

## 5.1 Remove the influence of bad measure

The default value of is set to , except for a few measures.

expe1 <- expe1 %>% mutate(lambda = 1) %>% filter(tps <=9)  
expe2 <- expe2 %>% mutate(lambda = 1)  
expe3 <- expe3 %>% mutate(lambda = 1)

A few observed values are far from the Gompertz curve fitted by minimizing least square.

change\_lambda1 <- tibble(  
 id = c(1, 3, 4, 6, 11), tps = c(8, 8, 8, 8, 8),  
 lambda = c(5, 5, 5, 5, 5)^2  
)  
change\_lambda2 <- tibble(id = c(3, 4), tps = c(0, 0), lambda = c(5, 5)^2)  
change\_lambda3 <- tibble(id = c(4, 11), tps = c(4, 4), lambda = c(5, 5)^2)  
expe1 <- expe1 %>% correct\_lambda(change\_lambda1)  
expe2 <- expe2 %>% correct\_lambda(change\_lambda2)  
expe3 <- expe3 %>% correct\_lambda(change\_lambda3)

## 5.2 Samples from the posterior distributions

We used Stan, see [<https://mc-stan.org>] and (Carpenter et al., 2017), to sample the posterior distribution.

bayes\_model <- stan\_model(file = here::here("stan/modele\_z0global.stan"))

To avoid been trapped around a low probability mode of the posterior, we start all MCMC algorithms with the least square estimates.

data1 <- set\_data(expe1)  
#> Groups have been coded as follow:  
#> Group number 1 is AA   
#> Group number 2 is 6-OHDA  
fit1 <- sampling(bayes\_model, data = data1,  
 chains = 4, iter = 10000,  
 init = function(...) init\_fun(par\_LS, data1,...),  
 refresh = 0,  
 control = list(adapt\_delta = 0.8, max\_treedepth = 10))

data2 <- set\_data(expe2)  
#> Groups have been coded as follow:  
#> Group number 1 is AA   
#> Group number 2 is 6-OHDA  
fit2 <- sampling(bayes\_model, data = data2,  
 chains = 4, iter = 10000,  
 init = function(...) init\_fun(par\_LS2, data2,...),  
 refresh = 0,  
 control = list(adapt\_delta = 0.8, max\_treedepth = 10))

data3 <- set\_data(expe3)  
#> Groups have been coded as follow:  
#> Group number 1 is AA   
#> Group number 2 is 6-OHDA  
fit3 <- sampling(bayes\_model, data = data3,  
 chains = 4, iter = 10000,  
 init = function(...) init\_fun(par\_LS3, data3,...),  
 refresh = 0,  
 control = list(adapt\_delta = 0.8, max\_treedepth = 10))

## 5.3 Reproduce Figures 9E and 9F of the paper

We can plot the credibility envelop of the growth curve using parameters drawn from the posterior distribution.

print(envelop(fit1, expe1 %>% filter(tps <= 9), id = 3, ci\_level = 0.5,  
 t = seq(0, 9, length.out = 101), color = "#000000"))

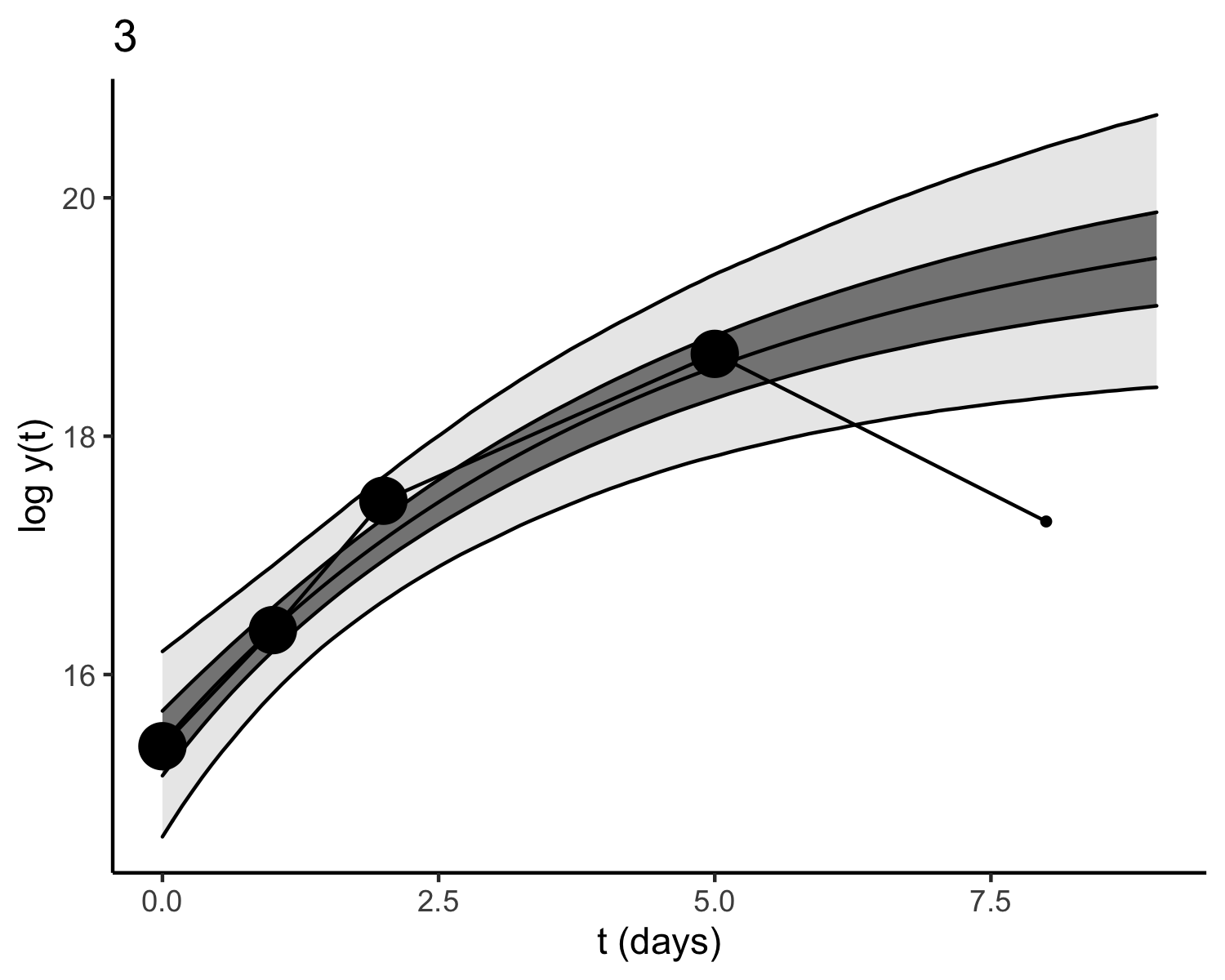


Figure 4.1: This is Fig. 9E of the original paper

print(envelop(fit1, expe1 %>% filter(tps <= 9), id = 8, ci\_level = 0.5,  
 t = seq(0, 9, length.out = 101), color = "#4B8EC7"))

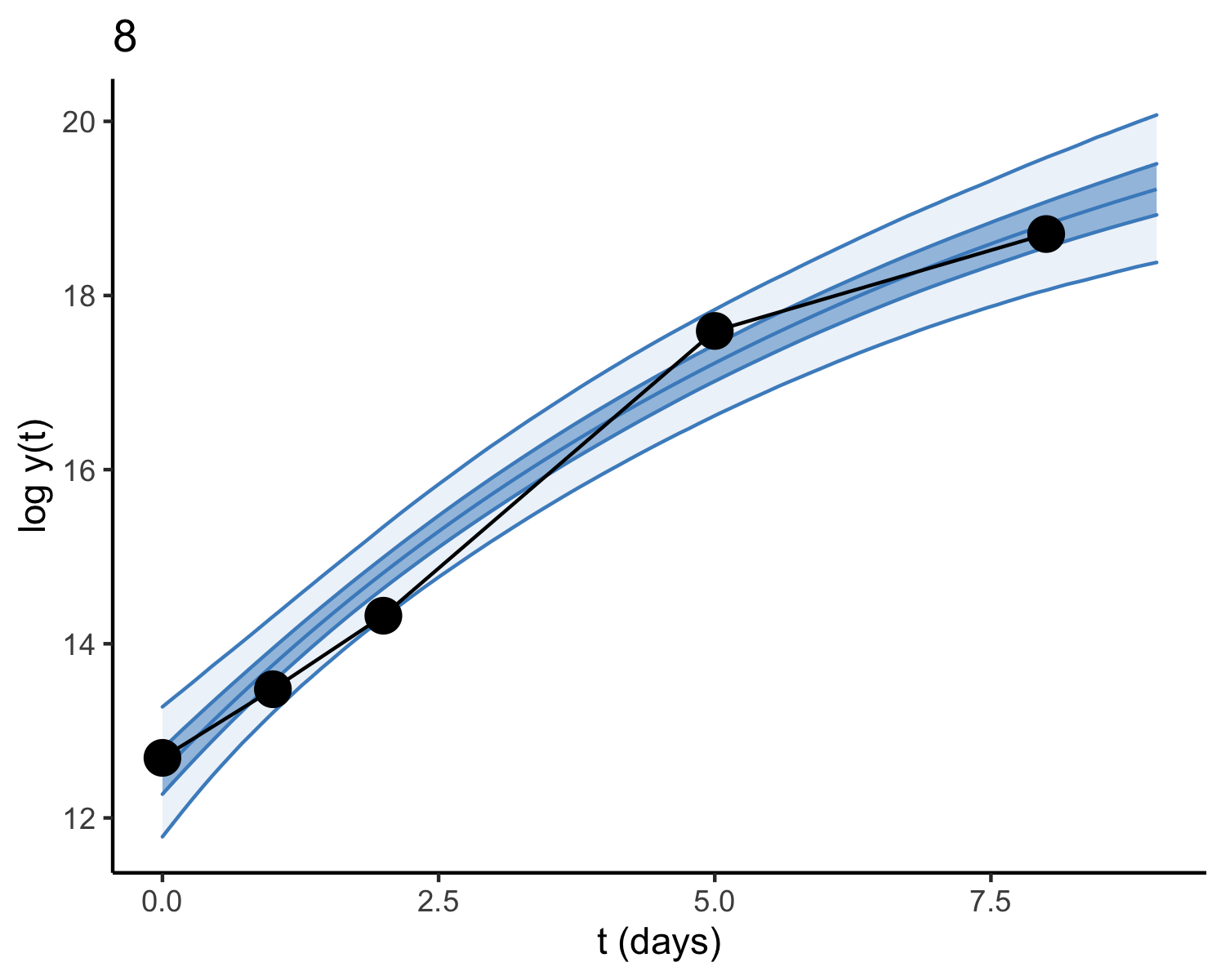


Figure 4.2: This is Fig. 9F of the original paper

## 5.4 Summarize the posterior samples

### 5.4.1 PK4A-Luc Syngeneic

mysummary(fit1, pars = c("z0bar", "tau0", "z0"))

| par | mean | se\_mean | sd | 2.5% | 25% | 50% | 75% | 97.5% |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| z0bar | 14.196 | 0.003 | 0.427 | 13.342 | 13.92 | 14.200 | 14.47 | 15.032 |
| tau0 | 1.306 | 0.002 | 0.292 | 0.865 | 1.10 | 1.261 | 1.46 | 1.995 |
| z0[1] | 12.979 | 0.003 | 0.395 | 12.196 | 12.71 | 12.980 | 13.25 | 13.745 |
| z0[2] | 15.446 | 0.003 | 0.399 | 14.666 | 15.18 | 15.446 | 15.72 | 16.218 |
| z0[3] | 15.422 | 0.003 | 0.400 | 14.639 | 15.15 | 15.426 | 15.70 | 16.194 |
| z0[4] | 14.260 | 0.003 | 0.385 | 13.498 | 14.00 | 14.265 | 14.52 | 15.000 |
| z0[5] | 14.892 | 0.003 | 0.382 | 14.132 | 14.64 | 14.896 | 15.15 | 15.636 |
| z0[6] | 15.514 | 0.003 | 0.388 | 14.743 | 15.26 | 15.517 | 15.78 | 16.266 |
| z0[7] | 15.574 | 0.003 | 0.385 | 14.799 | 15.32 | 15.583 | 15.83 | 16.326 |
| z0[8] | 12.533 | 0.003 | 0.384 | 11.784 | 12.27 | 12.533 | 12.79 | 13.276 |
| z0[9] | 12.451 | 0.003 | 0.380 | 11.699 | 12.20 | 12.449 | 12.71 | 13.191 |
| z0[10] | 12.996 | 0.003 | 0.391 | 12.217 | 12.73 | 13.000 | 13.27 | 13.753 |
| z0[11] | 14.363 | 0.002 | 0.392 | 13.591 | 14.10 | 14.369 | 14.62 | 15.120 |

mysummary(fit1, pars = c("log\_bbar", "taub", "log\_b"))

| par | mean | se\_mean | sd | 2.5% | 25% | 50% | 75% | 97.5% |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| log\_bbar[1] | 19.805 | 0.026 | 1.096 | 18.160 | 19.082 | 19.633 | 20.331 | 22.498 |
| log\_bbar[2] | 22.635 | 0.029 | 1.672 | 20.012 | 21.479 | 22.397 | 23.560 | 26.600 |
| taub[1] | 1.181 | 0.003 | 0.353 | 0.690 | 0.935 | 1.116 | 1.356 | 2.050 |
| taub[2] | 1.191 | 0.004 | 0.441 | 0.647 | 0.894 | 1.093 | 1.373 | 2.324 |
| log\_b[1] | 19.124 | 0.033 | 1.500 | 16.879 | 18.082 | 18.882 | 19.927 | 22.696 |
| log\_b[2] | 21.331 | 0.022 | 1.030 | 19.843 | 20.632 | 21.166 | 21.825 | 23.873 |
| log\_b[3] | 20.330 | 0.025 | 1.173 | 18.638 | 19.540 | 20.128 | 20.892 | 23.231 |
| log\_b[4] | 19.890 | 0.029 | 1.346 | 17.865 | 18.954 | 19.673 | 20.607 | 23.124 |
| log\_b[5] | 19.560 | 0.025 | 1.091 | 18.124 | 18.842 | 19.336 | 20.005 | 22.437 |
| log\_b[6] | 19.350 | 0.028 | 1.261 | 17.566 | 18.513 | 19.115 | 19.923 | 22.482 |
| log\_b[7] | 19.071 | 0.025 | 1.079 | 17.721 | 18.393 | 18.832 | 19.450 | 21.999 |
| log\_b[8] | 22.355 | 0.030 | 1.842 | 19.563 | 21.039 | 22.096 | 23.376 | 26.705 |
| log\_b[9] | 22.845 | 0.029 | 1.787 | 20.127 | 21.592 | 22.583 | 23.811 | 27.116 |
| log\_b[10] | 22.428 | 0.028 | 1.716 | 19.942 | 21.210 | 22.135 | 23.319 | 26.615 |
| log\_b[11] | 23.117 | 0.028 | 1.719 | 20.466 | 21.898 | 22.866 | 24.064 | 27.171 |

mysummary(fit1, pars = c("abar", "taua", "a"))

| par | mean | se\_mean | sd | 2.5% | 25% | 50% | 75% | 97.5% |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| abar[1] | 0.218 | 0.002 | 0.068 | 0.097 | 0.169 | 0.215 | 0.263 | 0.361 |
| abar[2] | 0.154 | 0.001 | 0.043 | 0.078 | 0.124 | 0.151 | 0.181 | 0.244 |
| taua[1] | 0.039 | 0.000 | 0.015 | 0.021 | 0.029 | 0.036 | 0.045 | 0.078 |
| taua[2] | 0.036 | 0.000 | 0.013 | 0.020 | 0.027 | 0.034 | 0.042 | 0.070 |
| a[1] | 0.204 | 0.001 | 0.073 | 0.083 | 0.150 | 0.198 | 0.251 | 0.363 |
| a[2] | 0.230 | 0.001 | 0.067 | 0.112 | 0.183 | 0.226 | 0.272 | 0.373 |
| a[3] | 0.231 | 0.001 | 0.075 | 0.101 | 0.178 | 0.225 | 0.278 | 0.391 |
| a[4] | 0.211 | 0.001 | 0.069 | 0.095 | 0.160 | 0.206 | 0.256 | 0.360 |
| a[5] | 0.220 | 0.002 | 0.076 | 0.084 | 0.165 | 0.216 | 0.269 | 0.380 |
| a[6] | 0.214 | 0.002 | 0.080 | 0.075 | 0.156 | 0.210 | 0.266 | 0.381 |
| a[7] | 0.218 | 0.002 | 0.083 | 0.067 | 0.159 | 0.215 | 0.271 | 0.393 |
| a[8] | 0.140 | 0.001 | 0.042 | 0.072 | 0.110 | 0.135 | 0.166 | 0.234 |
| a[9] | 0.144 | 0.001 | 0.039 | 0.079 | 0.117 | 0.141 | 0.168 | 0.230 |
| a[10] | 0.160 | 0.001 | 0.047 | 0.082 | 0.127 | 0.156 | 0.190 | 0.263 |
| a[11] | 0.172 | 0.001 | 0.047 | 0.092 | 0.138 | 0.168 | 0.201 | 0.274 |

### 5.4.2 R211-Luc Syneneic

mysummary(fit2, pars = c("z0bar", "tau0", "z0"))

| par | mean | se\_mean | sd | 2.5% | 25% | 50% | 75% | 97.5% |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| z0bar | 9.531 | 0.005 | 0.639 | 8.211 | 9.137 | 9.545 | 9.941 | 10.757 |
| tau0 | 1.536 | 0.004 | 0.467 | 0.870 | 1.211 | 1.457 | 1.776 | 2.650 |
| z0[1] | 11.142 | 0.003 | 0.438 | 10.282 | 10.848 | 11.145 | 11.439 | 11.993 |
| z0[2] | 10.296 | 0.004 | 0.463 | 9.376 | 9.989 | 10.300 | 10.612 | 11.187 |
| z0[3] | 6.582 | 0.011 | 0.988 | 4.446 | 5.964 | 6.650 | 7.269 | 8.342 |
| z0[4] | 8.559 | 0.006 | 0.719 | 7.059 | 8.099 | 8.583 | 9.055 | 9.902 |
| z0[5] | 10.792 | 0.002 | 0.409 | 9.986 | 10.520 | 10.796 | 11.066 | 11.592 |
| z0[6] | 9.267 | 0.003 | 0.399 | 8.486 | 8.993 | 9.274 | 9.537 | 10.044 |
| z0[7] | 10.001 | 0.002 | 0.399 | 9.213 | 9.734 | 10.000 | 10.270 | 10.783 |

mysummary(fit2, pars = c("log\_bbar", "taub", "log\_b"))

| par | mean | se\_mean | sd | 2.5% | 25% | 50% | 75% | 97.5% |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| log\_bbar[1] | 15.709 | 0.025 | 1.350 | 13.676 | 14.830 | 15.499 | 16.334 | 19.071 |
| log\_bbar[2] | 22.892 | 0.045 | 2.632 | 18.637 | 21.005 | 22.579 | 24.456 | 28.857 |
| taub[1] | 1.194 | 0.004 | 0.469 | 0.639 | 0.887 | 1.087 | 1.376 | 2.399 |
| taub[2] | 1.323 | 0.006 | 0.559 | 0.671 | 0.950 | 1.190 | 1.540 | 2.763 |
| log\_b[1] | 15.263 | 0.023 | 1.201 | 13.874 | 14.514 | 14.980 | 15.639 | 18.635 |
| log\_b[2] | 16.037 | 0.022 | 1.125 | 14.616 | 15.315 | 15.803 | 16.462 | 18.930 |
| log\_b[3] | 15.532 | 0.028 | 1.511 | 13.432 | 14.503 | 15.248 | 16.232 | 19.290 |
| log\_b[4] | 23.293 | 0.043 | 2.662 | 19.132 | 21.372 | 22.945 | 24.849 | 29.483 |
| log\_b[5] | 22.938 | 0.045 | 2.787 | 18.451 | 20.936 | 22.606 | 24.605 | 29.314 |
| log\_b[6] | 22.552 | 0.049 | 3.029 | 17.447 | 20.384 | 22.287 | 24.430 | 29.255 |
| log\_b[7] | 23.024 | 0.046 | 2.817 | 18.498 | 20.979 | 22.690 | 24.669 | 29.549 |

mysummary(fit2, pars = c("abar", "taua", "a"))

| par | mean | se\_mean | sd | 2.5% | 25% | 50% | 75% | 97.5% |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| abar[1] | 0.203 | 0.001 | 0.070 | 0.076 | 0.154 | 0.200 | 0.248 | 0.350 |
| abar[2] | 0.077 | 0.000 | 0.027 | 0.030 | 0.058 | 0.075 | 0.094 | 0.136 |
| taua[1] | 0.044 | 0.000 | 0.020 | 0.021 | 0.031 | 0.038 | 0.051 | 0.096 |
| taua[2] | 0.034 | 0.000 | 0.011 | 0.019 | 0.026 | 0.031 | 0.039 | 0.061 |
| a[1] | 0.207 | 0.001 | 0.083 | 0.062 | 0.148 | 0.202 | 0.259 | 0.385 |
| a[2] | 0.223 | 0.001 | 0.075 | 0.091 | 0.170 | 0.219 | 0.270 | 0.381 |
| a[3] | 0.180 | 0.001 | 0.060 | 0.079 | 0.136 | 0.175 | 0.217 | 0.313 |
| a[4] | 0.099 | 0.000 | 0.028 | 0.054 | 0.078 | 0.095 | 0.115 | 0.164 |
| a[5] | 0.074 | 0.000 | 0.024 | 0.038 | 0.056 | 0.070 | 0.087 | 0.131 |
| a[6] | 0.061 | 0.000 | 0.021 | 0.032 | 0.046 | 0.057 | 0.071 | 0.113 |
| a[7] | 0.075 | 0.000 | 0.023 | 0.041 | 0.059 | 0.071 | 0.088 | 0.130 |

### 5.4.3 R211-Luc Athymic nude

mysummary(fit3, pars = c("z0bar", "tau0", "z0"))

| par | mean | se\_mean | sd | 2.5% | 25% | 50% | 75% | 97.5% |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| z0bar | 12.866 | 0.003 | 0.369 | 12.129 | 12.63 | 12.868 | 13.102 | 13.594 |
| tau0 | 1.089 | 0.002 | 0.253 | 0.705 | 0.91 | 1.052 | 1.229 | 1.686 |
| z0[1] | 11.456 | 0.003 | 0.442 | 10.566 | 11.16 | 11.459 | 11.760 | 12.305 |
| z0[2] | 12.306 | 0.004 | 0.460 | 11.393 | 11.99 | 12.310 | 12.619 | 13.189 |
| z0[3] | 13.614 | 0.004 | 0.482 | 12.670 | 13.29 | 13.618 | 13.938 | 14.566 |
| z0[4] | 12.733 | 0.004 | 0.454 | 11.836 | 12.43 | 12.735 | 13.042 | 13.615 |
| z0[5] | 12.464 | 0.003 | 0.409 | 11.646 | 12.19 | 12.468 | 12.741 | 13.249 |
| z0[6] | 13.392 | 0.003 | 0.422 | 12.562 | 13.11 | 13.393 | 13.676 | 14.213 |
| z0[7] | 14.953 | 0.004 | 0.466 | 14.027 | 14.64 | 14.958 | 15.272 | 15.853 |
| z0[8] | 13.305 | 0.003 | 0.424 | 12.465 | 13.03 | 13.307 | 13.592 | 14.139 |
| z0[9] | 12.921 | 0.004 | 0.457 | 12.013 | 12.62 | 12.922 | 13.226 | 13.820 |
| z0[10] | 11.769 | 0.004 | 0.439 | 10.889 | 11.48 | 11.775 | 12.070 | 12.603 |
| z0[11] | 12.749 | 0.003 | 0.437 | 11.888 | 12.46 | 12.751 | 13.043 | 13.611 |

mysummary(fit3, pars = c("log\_bbar", "taub", "log\_b"))

| par | mean | se\_mean | sd | 2.5% | 25% | 50% | 75% | 97.5% |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| log\_bbar[1] | 18.206 | 0.022 | 1.239 | 15.973 | 17.369 | 18.129 | 18.96 | 20.858 |
| log\_bbar[2] | 19.743 | 0.035 | 1.543 | 17.210 | 18.639 | 19.603 | 20.67 | 23.204 |
| taub[1] | 1.585 | 0.008 | 0.623 | 0.733 | 1.154 | 1.485 | 1.88 | 3.124 |
| taub[2] | 1.498 | 0.008 | 0.570 | 0.702 | 1.076 | 1.413 | 1.80 | 2.859 |
| log\_b[1] | 15.962 | 0.044 | 2.065 | 13.752 | 14.426 | 15.058 | 17.32 | 20.860 |
| log\_b[2] | 18.639 | 0.019 | 1.004 | 17.210 | 17.945 | 18.451 | 19.13 | 21.070 |
| log\_b[3] | 19.191 | 0.012 | 0.719 | 18.132 | 18.714 | 19.085 | 19.52 | 20.935 |
| log\_b[4] | 18.810 | 0.016 | 0.896 | 17.556 | 18.217 | 18.647 | 19.20 | 21.085 |
| log\_b[5] | 18.699 | 0.047 | 2.223 | 15.468 | 16.829 | 18.481 | 20.23 | 23.480 |
| log\_b[6] | 19.917 | 0.033 | 1.608 | 17.536 | 18.736 | 19.664 | 20.84 | 23.708 |
| log\_b[7] | 20.776 | 0.024 | 1.228 | 19.028 | 19.921 | 20.554 | 21.39 | 23.760 |
| log\_b[8] | 20.678 | 0.029 | 1.491 | 18.418 | 19.613 | 20.459 | 21.50 | 24.184 |
| log\_b[9] | 19.990 | 0.028 | 1.381 | 18.050 | 19.017 | 19.729 | 20.69 | 23.427 |
| log\_b[10] | 17.819 | 0.060 | 2.740 | 14.186 | 15.295 | 17.481 | 19.99 | 23.309 |
| log\_b[11] | 20.282 | 0.031 | 1.490 | 18.100 | 19.211 | 20.034 | 21.09 | 23.836 |

mysummary(fit3, pars = c("abar", "taua", "a"))

| par | mean | se\_mean | sd | 2.5% | 25% | 50% | 75% | 97.5% |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| abar[1] | 0.256 | 0.002 | 0.083 | 0.106 | 0.196 | 0.254 | 0.312 | 0.427 |
| abar[2] | 0.139 | 0.001 | 0.050 | 0.060 | 0.102 | 0.133 | 0.171 | 0.251 |
| taua[1] | 0.056 | 0.000 | 0.031 | 0.022 | 0.035 | 0.047 | 0.068 | 0.136 |
| taua[2] | 0.044 | 0.000 | 0.017 | 0.022 | 0.032 | 0.041 | 0.052 | 0.089 |
| a[1] | 0.202 | 0.002 | 0.117 | 0.037 | 0.086 | 0.204 | 0.292 | 0.427 |
| a[2] | 0.253 | 0.001 | 0.080 | 0.117 | 0.195 | 0.248 | 0.305 | 0.421 |
| a[3] | 0.300 | 0.001 | 0.084 | 0.146 | 0.242 | 0.296 | 0.352 | 0.473 |
| a[4] | 0.273 | 0.001 | 0.085 | 0.122 | 0.212 | 0.270 | 0.327 | 0.450 |
| a[5] | 0.102 | 0.001 | 0.058 | 0.033 | 0.058 | 0.085 | 0.137 | 0.242 |
| a[6] | 0.134 | 0.001 | 0.051 | 0.057 | 0.095 | 0.125 | 0.165 | 0.253 |
| a[7] | 0.168 | 0.001 | 0.059 | 0.075 | 0.125 | 0.162 | 0.204 | 0.300 |
| a[8] | 0.137 | 0.001 | 0.043 | 0.068 | 0.105 | 0.131 | 0.163 | 0.236 |
| a[9] | 0.174 | 0.001 | 0.059 | 0.077 | 0.130 | 0.168 | 0.210 | 0.303 |
| a[10] | 0.105 | 0.002 | 0.079 | 0.020 | 0.041 | 0.075 | 0.160 | 0.286 |
| a[11] | 0.154 | 0.001 | 0.053 | 0.072 | 0.115 | 0.148 | 0.187 | 0.273 |

## 5.5 Represent the posterior distibutions

### 5.5.1 Reproduce Figure 9G

print(compare\_type(fit1, xlim = c(17, 31), ylim = c(0, 0.45)))

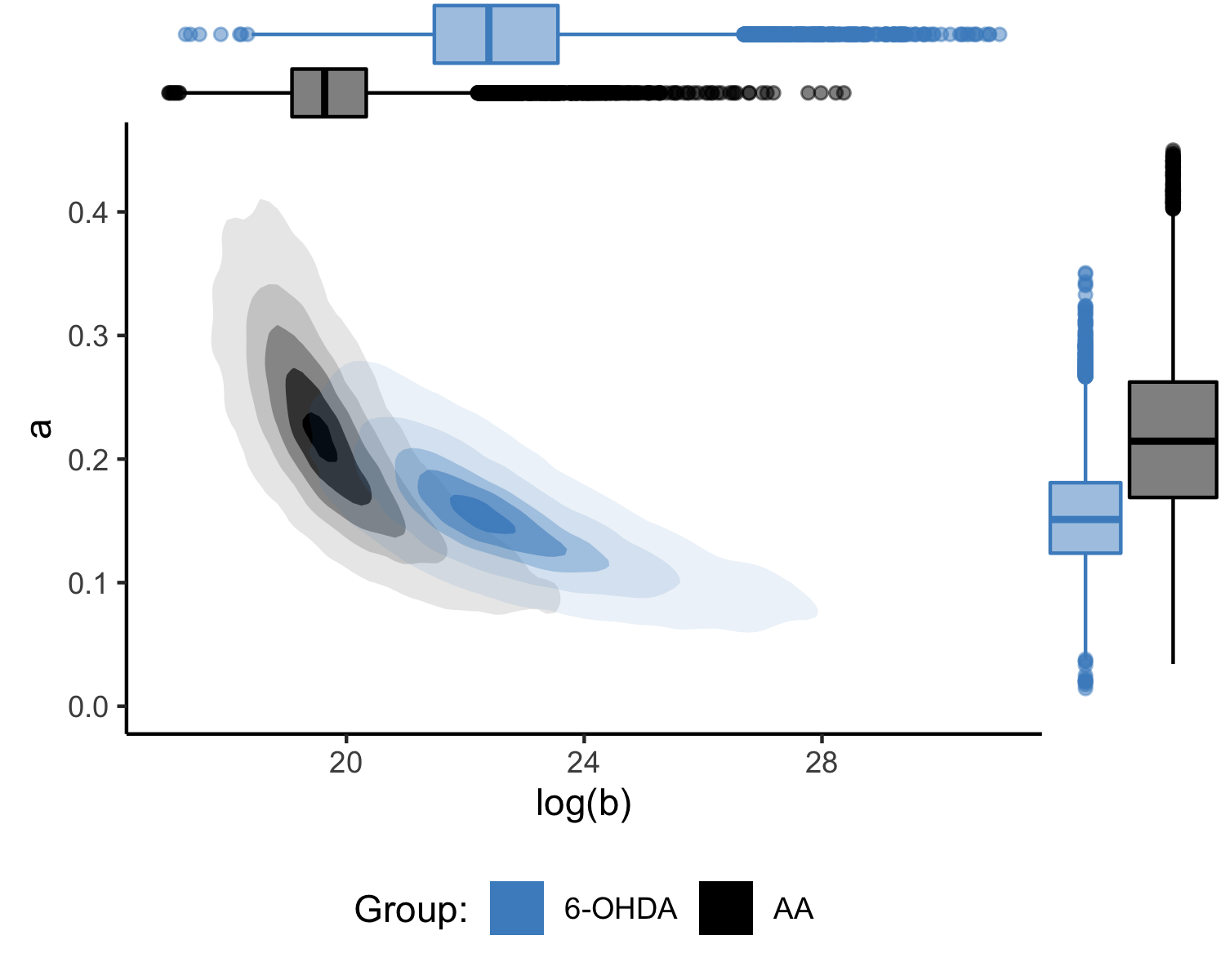


Figure 4.3: This is Fig. 9G of the original paper

The posterior probability that is:

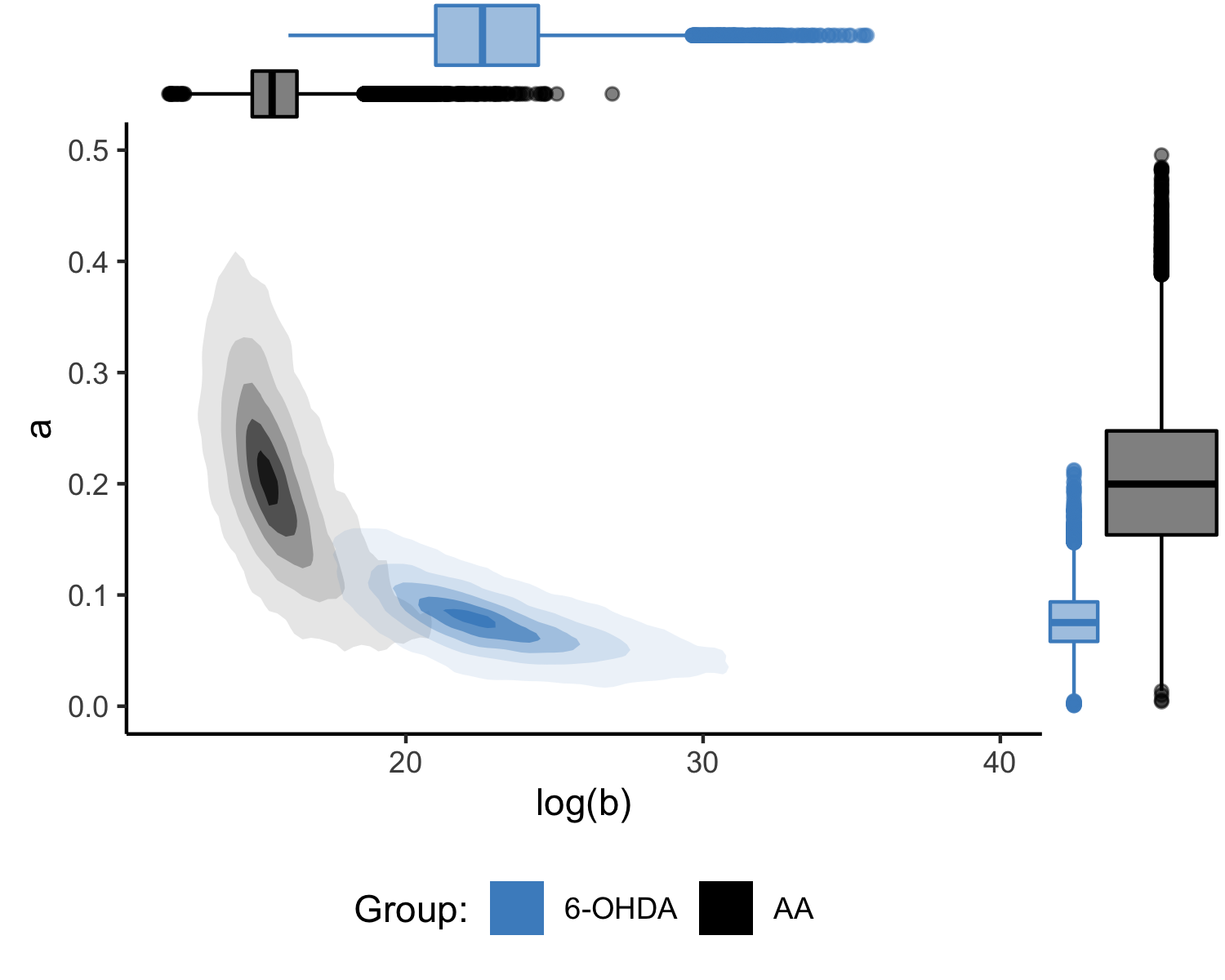
postdata1 <- as.data.frame(fit1, pars = c("abar", "log\_bbar"))  
mean(postdata1$`abar[1]` > postdata1$`abar[2]`)  
#> [1] 0.8544

The posterior probability that is:

mean(postdata1$`log\_bbar[1]` < postdata1$`log\_bbar[2]`)  
#> [1] 0.9589

### 5.5.2 Reproduce Figure 9H

print(compare\_type(fit2, xlim = c(12, 40), ylim = c(0, 0.5)))

 The posterior probability that is:

postdata2 <- as.data.frame(fit2, pars = c("abar", "log\_bbar"))  
mean(postdata2$`abar[1]` > postdata2$`abar[2]`)  
#> [1] 0.9686

The posterior probability that is:

mean(postdata2$`log\_bbar[1]` < postdata2$`log\_bbar[2]`)  
#> [1] 0.9959

### 5.5.3 Reproduce Figure 9I

compare\_type(fit3, xlim = c(12, 40), ylim = c(0, 0.5))

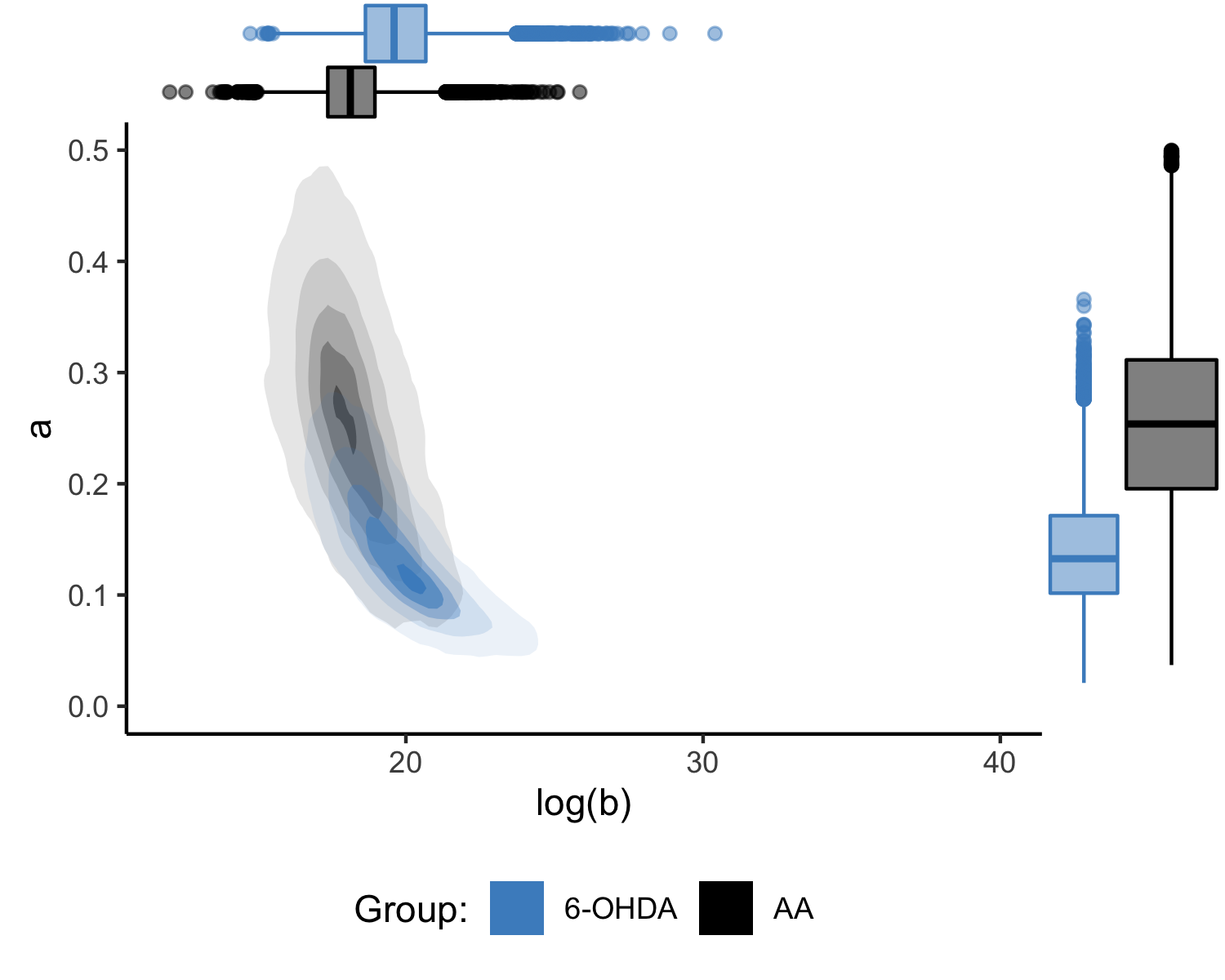


Figure 4.5: This is Fig. 9I of the original paper

The posterior probability that is:

postdata3 <- as.data.frame(fit3, pars = c("abar", "log\_bbar"))  
mean(postdata3$`abar[1]` > postdata3$`abar[2]`)  
#> [1] 0.8568

The posterior probability that is:

mean(postdata3$`log\_bbar[1]` < postdata3$`log\_bbar[2]`)  
#> [1] 0.7648

# 6 References

Carpenter, B., Gelman, A., Hoffman, M. D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., & Riddell, A. (2017). Stan: A probabilistic programming language. *Journal of Statistical Software*, *76*(1).

Gelman, A., & Hill, J. (2006). *Data analysis using regression and multilevel/hierarchical models*. Cambridge university press.

Hartung, N., Mollard, S., Barbolosi, D., Benabdallah, A., Chapuisat, G., Henry, G., Giacometti, S., Iliadis, A., Ciccolini, J., Faivre, C.others. (2014). Mathematical modeling of tumor growth and metastatic spreading: Validation in tumor-bearing mice. *Cancer Research*, *74*(22), 6397–6407.

Lavielle, M. (2014). *Mixed effects models for the population approach: Models, tasks, methods and tools*. CRC press.